

## Assignment: Implementing and Comparing Gradient Descent and ID3

### Intro to Machine Learning

### Objective

The goal of this assignment is to:

1. Understand and implement a single-layer neural network using gradient descent.
2. Understand and implement the ID3 algorithm for decision trees.
3. Compare the performance of these algorithms on the same dataset.

### Instructions

#### Part 1: Single-Layer Neural Network with Gradient Descent

1. **Implementation:**

- Implement a single-layer neural network with sigmoid activation in Python.
- Implement and use gradient descent to minimize the error with cross-entropy loss.
- Compute the gradients (no use of libraries like PyTorch or TensorFlow).
- Initialize weights and biases randomly. Think about different distributions such as normal and uniform. You are encouraged to read online about common initializations.
- Try different learning rates and weights initializations.
- Train with Gradient descent until a stopping condition (of your choice) is reached. Examples for stopping conditions: goal accuracy, reaching convergence, fixed number of epochs.
- **Output:**
- In the final report include the results for the best learning rate and initialization, mention some other options that you've tried and briefly explain why you chose the ones that you did (in what sense they were the best).
- Plot the loss curves (train and test) over epochs.
- Briefly analyse the plots, do they match the theory?

- Provide the final weights and biases.
- Explain your choice of stopping condition.

## Part 2: ID3 Algorithm

### 1. Implementation:

- Implement the ID3 algorithm for constructing a decision tree.
- Use information gain with entropy to determine the best feature to split the dataset at each node.
- Allow the algorithm to handle categorical and numerical data.

### 2. Output:

- Visualize the constructed tree (using any suitable Python library or by printing the tree structure).

## Part 3: Comparison

### 1. Dataset:

- Use Wisconsin breast cancer dataset from sklearn  
[https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load\\_breast\\_cancer.html](https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_breast_cancer.html)

### 2. Tasks:

- Load the dataset and split it to 80% train and 20% test.
- Train both the neural network and the decision tree on the dataset.
- Compare the accuracy (percent of the correct predictions out of all of the predictions) models.
- Discuss the strengths and weaknesses of each algorithm based on your findings.

### 3. Submission:

- Include:
  - Python code for both implementations.
  - A report discussing your methodology, results, and comparison.
  - Plots and visualizations to support your findings.

## Guidelines

- **Code Quality:** Write clean and modular code. Use comments to explain your logic.
- **Documentation:** Clearly explain your implementation choices and assumptions in the report.

- **Evaluation:** Evaluate your models on test and train datasets and discuss overfitting/underfitting.
- **Tools:** Use standard Python libraries such as numpy, matplotlib, and sklearn- for dataset handling only, (the algorithms beside data loading shouldn't use sklearn).

## Submission Format

- The following files, unzipped:
  1. Python (.py format, no notebooks) script(s) for the neural network and ID3 algorithm.
  2. A report in PDF format, including the graphs.
- On the top of each file write the names and IDs of the students in your group. beware: you'll lose points for not doing so.

## Deadlines and Grading

- **Submission Deadline:** 16.1.25
- **Grading:**
  - Neural Network Implementation: 40%
  - ID3 Implementation: 40%
  - Comparison and Report: 20%
  - The grading will take into consideration: The consistency of the implementation with the guidelines and what we learned in the course, Efficiency (redundant loops etc.), order and legibility.

---

**Note:** This assignment is designed to deepen your understanding of the basic building blocks of machine learning algorithms. You are allowed to get some assistance from online resources, but plagiarism will not be tolerated and teach you nothing. That includes ChatGPT. Ensure your work is original and includes proper citations for any external resources used.